A LIBRARY OF SOFTWARE FOR THE **IBM PC** & **PCjr**

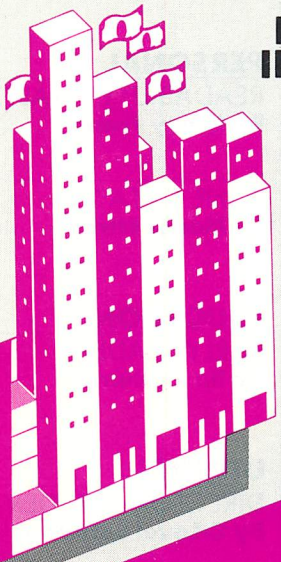
PC DISK MAGAZINE

VOL. 1 NO. 6

$34.95
DISKETTE INCLUDED

USER MANUAL

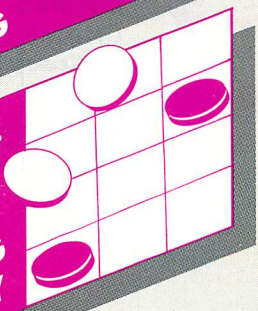
READABILITY ANALYZER: DETERMINE THE GRADE LEVEL OF YOUR PROSE



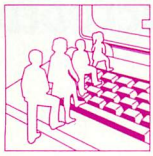
STOCK MINDER: COMPREHENSIVE WALL STREET TRACKING SYSTEM

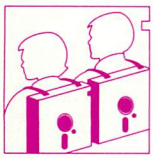
EARLY READER: COLORFUL GRAPHIC DISPLAYS DESIGNED TO AID LEARNING

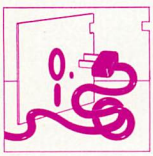
CHAMELEON: INSCRUTABLE GAME OF OUTFLANKING STRATEGY



Contents







DEPARTMENTS

TERMS AND CONDITIONS

LIMITED WARRANTY

In the unlikely event that the disk contained in this issue is defective in manufacture, return it within 30 days after purchase or receipt to *PC Disk Magazine,* P.O. Box 5930, Cherry Hill, NJ 08034, and it will be replaced at no charge. Merchandise for exchange must be accompanied by proof of purchase or a subscription label.

The publisher shall have no liability or responsibility to any purchaser or subscriber or to any other person or entity with respect to any liability, loss or damage directly or indirectly arising out of the use of the disk and the programs contained in this issue, including, but not limited to, any loss of business or any other incidental or consequential damages.

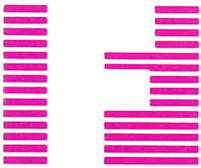
COPYRIGHT

A LIBRARY OF SOFTWARE FOR THE **IBM PC** & **PCjr**

PC DISK MAGAZINE

PC Disk Magazine is published 9 times per year, by Ziff-Davis Publishing Company, One Park Avenue, New York, N.Y. 10016. Subscriptions are available at $149.00 for 6 issues. POSTMASTER: send address changes to *PC Disk Magazine*, P.O. Box 5930, Cherry Hill, NJ 08034
Editorial and Business Office: One Park Ave., New York, NY 10016. Editorial (212) 725-7770. For subscription inquiries and service write to *PC Disk Magazine*, P.O. Box 5930, Cherry Hill, NJ 08034

ISBN #087165-106-8 PC DISK MAGAZINE VOLUME 1 NUMBER 6



FROM THE EDITOR

Microcomputer software has become one of the fastest growing industries in existence today. The reason for this is fairly obvious: a microcomputer, with its vast range of possible uses, is only as good as the software that exists for it. Software is ultimately the driving force behind our continuously expanding use of these machines.

The marketplace today is awash with an incredible number of microcomputer software products, and dozens more arrive each day. The crucial question in assessing this computer cornucopia is: What do people want to do with a microcomputer? This broad question breaks down into three component questions: 1) What practical functions can a computer be used for, 2) How do people want to work with a computer, and 3) What's the cost/value balance for doing things with a computer.

We all know that there's a lot of business software available for microcomputers. That's great for companies, but what about individuals interested in using computers for either personal or professional purposes? Well, there are plenty of games, and educational software is booming, but what about that loan analyzer you need, or the readability assessor you want—programs that do specific, practical jobs and don't cost a fortune? The bottom line is that there is not a lot of software to serve your *particular* personal needs.

Some would argue that there's a lot of general purpose software available which you can tailor to your needs. But does everyone really want to immerse themselves in the technology in order to use the tool? Some people do, but I think many more do not. Many people would like to treat a computer like any other appliance—a device that stands ready to perform a task quickly and easily. It just happens that this one device can be used for many tasks.

Then there's the cost factor. Why spend $200 or more for a spreadsheet program when all you want to do is a loan amortization schedule? At a point you may have enough uses for such a package to justify its cost. But the same will not be true for all the particular things you can use your microcomputer for.

No doubt you've guessed my punchline. *PC Disk Magazine* offers a collection of fully functional programs in every issue, to do those various jobs for which you bought your microcomputer. Hopefully, we'll give you some new uses for your machine as well. We offer programs for those who simply want to start up and go, and for those who want to get into the computer (in addition to our utilities, source code can be viewed, modified, and cannibalized in many cases). Best of all, the price for this substantial collection represents a great value. Need I say more?

UBMISSION PROCEDURE

What sets *PC Disk Magazine* apart from most other publications is that we want and need your direct involvement. That's why we developed the "Software Submission Plan." If you're interested in submitting software you've developed either on your own or with others, let us know, and we'll send you the Submission Plan booklet. We would like to give you an outline of our Submission Plan here in order to stimulate your imagination and your interest.

The Software Submission Plan provides you with an opportunity to profit directly from your work. Under the plan, *PC Disk Magazine* will pay you a royalty on every issue sold, containing your software. Best of all, this opportunity comes without elaborate restrictions. Our desire is to license software for publication only for the disk magazine format.

To explore the considerable opportunities for publication in *PC Disk Magazine*, write to us and ask for a Software Submission Plan booklet. The address is:

PC Disk Magazine
Author Submissions
One Park Avenue
New York, N.Y. 10016

Once you've received the Submission Plan, read it over carefully. If you feel your software fits the requirements set out in the plan, complete the enclosed Software Submission Agreement and return it to us. Do not send us any program code or documentation when you return the Submission Agreement.

Following receipt of the Submission Agreement, the editors of *PC Disk Magazine* will evaluate the submission. At that point, we'll either indicate a lack of interest in the software or, if we are interested, we'll issue a submission authorization number and ask you to provide us with an executable copy of the program along with documentation on 5¼" diskettes. Again, do not send us any additional materials until you receive a submission authorization number.

If we subsequently decide that we would like to publish the software in *PC Disk Magazine*, we will offer a Software Contract, which will include such items as royalties, advances, and program and documentation changes required (if any). You will be asked to complete and test any program modifications agreed to in the Software Contract, and *PC Disk Magazine* will conduct a formal validation of the program and documentation.

We estimate that this process, from our initial evaluation to publication, takes approximately 3 to 6 months. This estimate is dependent upon a number of factors, and the process may take more or less time for your submission.

We look forward to hearing from you.

TECHNICAL PREFACE

To help our readers make the most of PC Disk Magazine, *we would like to provide some background information concerning the editorial diskette, the accompanying manual, and how to use both. We don't expect all of the following topics to be of interest to all our readers. Nevertheless we prefer to err on the side of comprehensive support, rather than leave any of our readers confused or bewildered. So we encourage everyone to at least skim this section to assure a solid background for the use of* PC Disk Magazine.

USAGE REQUIREMENTS

PC Disk Magazine has been designed for use on an IBM Personal Computer with a minimal set of hardware components: a keyboard, a monitor, and the computer itself. Running DOS 1.1, a minimum of 64K of main memory is required. Under DOS 2.0 and DOS 2.1, a minimum of 128K is required. The display unit can be a monochrome display adapter and monitor, or the color graphics display adapter with either a color monitor, a black and white monitor, or an RF Modulator and TV set. The computer itself can be the PC, the PC-XT or the PCjr. For the PCjr we recommend the use of a monitor rather than a TV set, since most of our software is written for an 80-column display. However a TV set does provide a workable display.

These three pieces of equipment are all you need to run the majority of *PC Disk Magazine* software. Wherever possible we try to make the use of any other hardware optional. So, for example, many of the programs will generate printed output, but a printer is not required to use them. Occasionally, however, due to the nature of a program or its design, a particular piece of equipment will be necessary. When a program requires a piece of equipment not in

the minimum configuration stated above, this component will be listed as a "Special Requirement" on the program's title page in this manual.

In regard to software, all *PC Disk Magazine* programs are designed to run under DOS 1.1, DOS 2.0 and DOS 2.1. Furthermore, all BASIC programs in the magazine are designed to run under Microsoft's Advanced BASIC. Neither DOS nor Advanced BASIC are provided on the *PC Disk Magazine* diskette; they must be acquired separately. As a rule, these are the only outside software elements you will need to use *PC Disk Magazine*. We will occasionally publish a program which uses some additional, publicly available software product. Any such additional software will be listed as a "Special Requirement" on the program's title page in this manual.

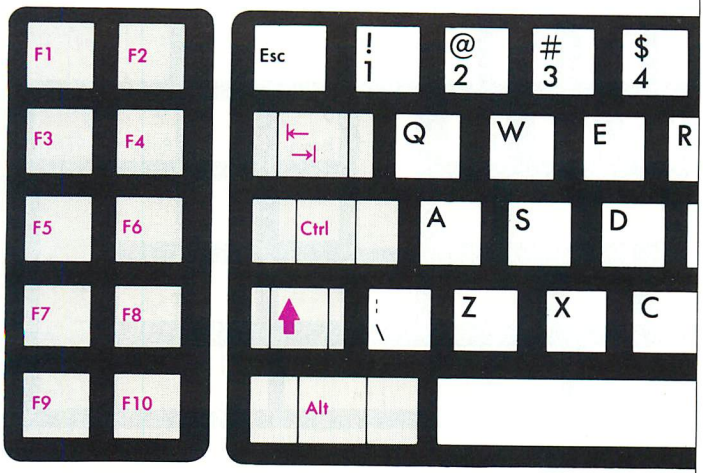
A closing remark on this topic is not so much a requirement as a recommendation. We recommend that you make a copy of your *PC Disk Magazine* diskette to work with, and save the original as a backup. In some cases, you will have to make a copy of the program in order to use it. The reason is that some programs create additional files as they run, and these files must be stored on diskette as well. You may have noticed that your *PC Disk Magazine* diskette is write-protected. Thus it cannot receive these additional files. So a separate, working copy is needed. These situations will be explicitly mentioned in the manual. In general though, where the manual refers to "your *PC Disk Magazine* diskette" you should read "your working copy of the *PC Disk Magazine* diskette."

THE IBM PC KEYBOARD

In *PC Disk Magazine* we have tried to make our instructions as clear as possible by the consistent use of special key symbols. In addition to all the common typewriter keys, which we print as they would appear when typed, the IBM PC keyboard has a number of special keys. We have designed symbols for these keys, which are intended to resemble as much as possible the keys themselves. Since these symbols are used extensively throughout the instructions, we felt the following road map and glossary would help you, our reader, get any needed bearings.

ATTENTION PCjr OWNERS:

The instructions in our manual are written for the PC keyboard. Section 4 of your "Guide to Operations" manual for the IBM PCjr provides complete information on how to translate PC keystrokes to PCjr keystrokes. Please refer to this section for guidance in following our operating instructions.



F6 THE FUNCTION KEYS

There are ten special keys called function keys located at the far left of the keyboard. They are numbered from F1 to F10. This stands for Function One, Function Two etc. These keys are often used to make single keystroke choices or commands.

Esc THE ESCAPE KEY

The Escape key is used most often for exactly what its name implies, to escape (exit) from various functions and processes.

Ctrl THE CONTROL KEY

This key is always used in conjunction with another key by pressing this key and the other key simultaneously. The purposes of the Control key vary widely depending on the application program.

|← →| THE TAB KEY

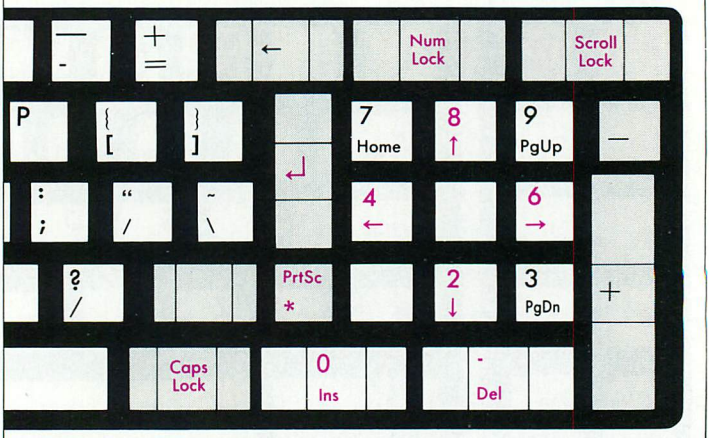
This key is commonly used for horizontal tabbing in text files. It is sometimes used by programs to allow rapid cursor movement during full-screen data entry.

← THE BACKSPACE KEY

The Backspace key is used to correct typing errors. By simply pressing the key, the preceding character is erased and a new character can be entered.

↑ THE SHIFT KEY

The Shift key is actually located on each side of the keyboard. It is used in conjunction with other keys to capitalize letters, get special symbols like: ! @ # $ % * () and other special functions.



PrtSc * THE PRINT SCREEN KEY

This key is used with the Shift key to get a printout of exactly what is on the screen. In computer lingo this is called a screen dump, a dump of all the information on the screen to the printer. In *PC Disk Magazine* we also refer to this capability as "The IBM Print Screen Facility."

↵ THE ENTER KEY

This is the most frequently used key on the keyboard. Almost every time you need to give information to the computer, you have to press this key to ENTER that information. This key can also be thought of as the carriage return, since it works similarly to the RETURN key on a typewriter.

Num Lock THE NUM LOCK KEY

This key determines whether the numeric keypad will print numbers or act as cursor movement controls. You can tell that NUM LOCK is on by pressing any of the number keys to see what is displayed on your screen. If it is the number, then the NUM LOCK is off. If the number is not displayed, then NUM LOCK is on. To change the setting press NUM LOCK once.

2 ↓ 4 ← 6 → 8 ↑ THE CURSOR CONTROL KEYS

These are the arrows that point up, down, and to each side. The NUM LOCK has to be on for these keys to be functional. These keys control cursor movement within some *PC DISK MAGAZINE* programs. They will move the cursor in the direction of the arrow.

0 Ins • Del THE INSERT AND DELETE KEYS

These keys really mean the INSERT and DELETE keys. And that is exactly how they are used. INS is used to insert new information and DEL is used to delete unwanted information. They are commonly used when editing BASIC programs, and can often be used when running BASIC programs as well.

Caps Lock THE CAPS LOCK KEY

This key is used to save you from having to hold the shift key down all the time to get capital letters.

Ctrl Scroll Lock THE CONTROL AND SCROLL LOCK KEYS

This key combination deserves special mention because of its importance in BASIC, the language of most *PC DISK MAGAZINE* software. These keys used together will interrupt the processing of any BASIC program. The keys should be used with caution because some interruptions can require you to start an entire procedure from the beginning.

TERMINOLOGY

In the preceding section we identified the special key symbols used in this manual, and gave a name to each one. For example:

is called the Enter key. In our instructional narrative, it sometimes makes more sense to refer to a special key by its name rather than its symbol. Thus the key names in the preceding section are also special terms for the purposes of this manual. Familiarize yourself with the names to facilitate your use of the manual, and refer to the preceding section as a glossary of key names when necessary.

In addition to the key names, a few other terms and phrases are used in this manual that may be unfamiliar to you.

We commonly speak of putting a diskette in the "default drive." This may seem like a needlessly vague phrase. After all, we know a diskette drive always has a one letter identifier associated with it, so why not refer explicitly to that letter? The problem with using an explicit letter reference is that it can create confusion about what exactly you must do. In other words, operationally it does not matter whether you put the diskette in the A Drive, the B Drive or even the C Drive (if you have a third diskette drive). What matters is that you put the diskette *in the drive that is currently active*, i.e. the drive whose letter prompt currently appears on the screen. This is your "default drive" because any disk command without a drive letter will look at the diskette in this active drive. So when you put a diskette in the "default drive," you can then issue commands referencing that diskette without the use of letter identifiers.

Every start-up procedure for a BASIC program requires you to "Load Advanced BASIC into your PC." To run a *PC Disk Magazine* BASIC program, the BASIC Interpreter must be up and running on your machine—you must be "in BASIC." BASIC is really a program like any other. To start it you must load it from a disk (or a cartridge in the case of PCjr) into your computer and start it running. This is

precisely what happens when you put your DOS diskette (or any diskette with the file BASICA.COM) in the default drive and type:

BASICA ↵

By so doing you "Load Advanced BASIC into your PC."

TEXT CONVENTIONS

Most of the textual conventions of this manual are fairly obvious. The use of special key symbols has been covered. The use of special key names in the narrative text has been discussed. That leaves two brief additional remarks concerning command lines.

The lines set apart from the narrative text, in bold print and a different color, are commands that should be typed in exactly as they appear. When two key symbols appear immediately next to each other in such a command line, they should be pressed simultaneously. For example:

means press the Shift key and the Print Screen key simultaneously, thereby printing a copy of the current screen on your printer.

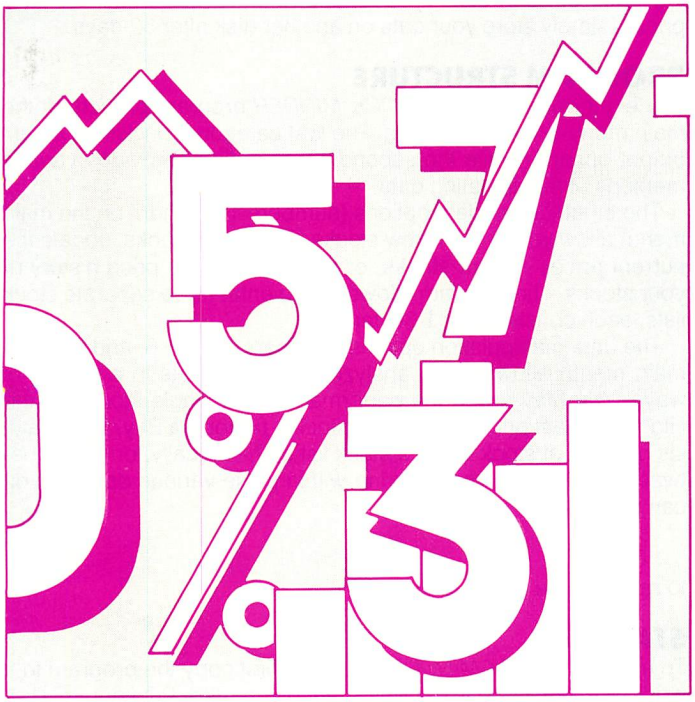
There is one exception to typing in command lines exactly as they appear. When a command includes a phrase such as "somename" or "programname" or "yourfile" you should replace that phrase (but not any punctuation) with a valid file name of your choice when you enter the command.

CALLING ALL PROGRAMMERS!!

If you have written software for the IBM PC that you feel would be of use or interest to others, *PC Disk Magazine* would like to know about it. Publishing is no longer just for poets and prosaists, but now it's for programmers too. Ask for our Software Submission Plan by writing to:

PC Disk Magazine
Author Submissions
One Park Avenue
New York, N.Y. 10016

BUSINESS



STOCK MINDER

By Ron Lashley

Special Requirements: Printer optional.
Files Used: SM.BAS

The Dow Jones was up again today . . . "It was another day of fierce trading . . . " Nowadays, the news is so full of reports on financial activity and changing market trends that it is usually difficult to keep track of your investments. Don't you wish there were an easier way of performing the unwieldy and time-consuming tasks of updating stock prices and analyzing whether to buy or sell? Well, there is. STOCK MINDER *is here to help you, the investor, use your knowledge of the market to organize your investments and to save yourself hours of work.*

BACKGROUND

STOCK MINDER stores and manipulates information for as many as three lists of stocks, each of which may contain up to 30 stocks

each. Three separate lists are used so that you can keep a different list for each of three different stock markets. This arrangement facilitates tracking a large number of stocks.

KEEP A COPY OF STOCK MINDER ON ANOTHER DISK

You may want to keep a copy of *STOCK MINDER* on another disk so that you can store it with your data. *STOCK MINDER* allows room for 52 price entries per stock, which lets you enter the weekly closing price of each stock you're following. To keep track of daily prices, simply store your data on another disk after 52 days.

PROGRAM STRUCTURE

The eight options of the *STOCK MINDER* program branch off the main menu in two categories. The first category consists of input/output options, while the second set of options provides various methods of manipulating data.

The input/output data options (numbers 1, 2, and 7 on the main menu) allow you to input new stocks, delete old stocks, update the current prices of your stocks, or print a complete price history of your stocks. The program allows you to enter three separate stock lists, each consisting of 1-30 items.

The data manipulation options (numbers 3, 4, 5, 6, and 8 on the main menu) allow you to analyze your stock lists in a variety of ways. You may graph the performance of a single stock, list the High-Low-Last prices for a list of stocks, perform a Buy/Sell analysis on a list of stocks, sort a stock list alphabetically, or switch between stock lists. When moving within these various options, you can press:

to return to the main menu.

START-UP

To execute *STOCK MINDER,* you must first copy the program to a work disk, where your stock lists can be created. Put your *PC Disk Magazine* diskette in the default drive and type:

```
COPY SM.BAS B: ↵
```

where B: is the drive containing a formatted work diskette. Next, load Advanced BASIC into your PC by putting a system disk in your default drive and type:

```
BASICA ↵
```

Then insert your work copy of *STOCK MINDER* in the default drive and type:

```
RUN "SM ↵
```

The title screen will then appear and query "Which stock list do you want (1, 2, or 3)?" Enter the appropriate number and the main menu will automatically appear.

THE MAIN MENU

In order to add a new stock name, first press:

```
1
```

This will cause the Add/Delete screen to appear. To add new stocks, press:

A ↵

Next, type the name of your stock and, in order to keep things organized, include in parentheses a one-letter code for the exchange. For example, you could use "(N)" to represent the New York Stock Exchange. Continue this process for all your stocks. After your last entry, press:

↵

to signal to the program that you're done. Remember that to facilitate keeping track of your stocks, you may want to keep a separate list for each exchange. To switch between stock lists, enter:

8

from the main menu and then type the number of the list you wish to view (1, 2, or 3).

KEEP A SEPARATE LIST FOR EACH EXCHANGE

One of three possible stock listings (with main menu).

To delete a stock, press:

1

from the main menu and then

D ↵

Enter the number of the stock to be deleted. Continue entering stock numbers until all desired deletions are made. To stop deleting stocks, press the Enter key.

To update a stock price, go to the main menu and press:

2

You will be prompted to enter the date. If there are existing prices for the date entered, they may be edited by selecting the stock number. At this point, you may enter data for a stock split. Enter a split in the following manner. If the split is 3 for 2, type:

3,2

When data for a stock split is entered, prior historical prices are automatically calculated.

If there are no existing prices for the date you have entered, the system will query, "Add Data (Y/N)?" If you enter a "Y", you will be prompted to enter a new price for each stock.

To graph the history of your stock, press:

3

from the main menu. The program asks you to type in the number of the stock you wish to see graphed and then displays the graph screen. Beneath the graph are the choices for your next action. Press:

G

to display another graph, press:

Esc

to go back to the main menu, or press:

↑ PrtSc *

to print your graph. Do not try to use a function key to print graphs, as this may cause the system to lock up.

The next option provides, at a glance, the highest, lowest, and the most recent prices you have entered for each stock. If you choose:

4

REVIEW YOUR STOCK'S HIGH-LOW-LAST PRICES

A listing of High-Low-Last prices

from the main menu, you can evaluate the present price of your stocks in relation to their history. To return to the main menu, press:

Esc

A Buy/Sell analysis is obtained by entering:

5

from the main menu.

This powerful routine prints the slope and the mean of the stock prices and the standard deviation of the last ten entries. A Buy or Sell decision appears in color (if it is available) or in highlighted characters (on a monochrome monitor). This information, combined with your knowledge about the quality of each stock and the number of shares you can afford, will guide your decision-making.

Analysis for a Delta also appears, giving you a relative indication of the volatility of each stock. The higher the number, the greater the price movement as a percentage of the average price.

After you enter and delete stocks over an extended period, the alphabetical order and the order by market (if different exchanges are on the same list) will become mixed up. By pressing:

6

from the main menu you can activate the sort routine and put the stocks in alphabetical order. They will also be arranged by market, provided that you have used market codes as suggested in option 1.

THE SORT ROUTINE WILL PUT THE STOCKS IN ALPHABETIC ORDER

To print a complete stock history, press:

7

from the main menu. You can print stock price history with four options: Single Stock, Stock List, Stock Range, and All. To print a listing for a single stock (say stock number 1), enter:

1 ↵

To print a list of stocks (say numbers 1, 2, and 4), enter:

1, 2, 4 ↵

To print a range of stocks (say 1-10), enter,

1-10 ↵

To print the price history of all the stocks, press:

↵

To exercise this price history option you obviously must have a printer.

EXITING

To exit the program from the main menu, press:

Esc

This returns you to BASIC.

UBLISHING PHILOSOPHY

A quick look at the editorial diskette for this issue of *PC Disk Magazine* reveals a fairly radical editorial decision: all ten BASIC programs have been published in original source code. Consequently, these programs can be listed and copied in their entirety. We at *PC Disk Magazine* choose neither to copy-protect our offerings, nor to make our program code inaccessible—a rather sharp departure from traditional software distribution.

WHY?

The most important reason behind our decision is our desire to make *PC Disk Magazine* of the greatest possible use to you. This desire involves several considerations. First, we want you to be able to learn from these programs. To do so you must be able to study the source code in order to understand the design concepts and programming techniques employed. Second, we want you to be able to adapt these programs to your own needs. You may choose to modify some programs, or expand them, or include them in programs of your own. Once again, you need the source code to do this. The third consideration is that to have the flexibility to use *PC Disk Magazine* most extensively, to modify and expand programs, to create different versions and to experiment, you must be able to make copies of *PC Disk Magazine* materials. Thus, our objective—to establish the closest possible relationship between this magazine and its readers—was the most compelling reason to publish copy-able source code.

Another reason worth mentioning is space. The same program requires 10 to 50 percent less storage space on our editorial diskette in source code than it would in compiled BASIC. So publishing programs in source code lets us give you more software in the fixed amount of diskette space available.

Now that you understand the reasons for our decision, we hope you will minimize our risk by honoring the legal copy restrictions that apply to *PC Disk Magazine*. Use the magazine as extensively as you like for yourself, but do not give listings or copies of our software materials to others. Be aware that the contents of *PC Disk Magazine* are copyrighted private property. Your technical freedom to copy these materials implies no legal right to distribute them. We ask that you act responsibly in your use of *PC Disk Magazine* and not abuse the spirit of free exchange. For our part, we will continue to make our material as useful to you as possible, with the expectation that, properly understood, this policy will best serve you our readers.

BUSINESS

FUNCTION PLOT

By Hugh A. Calvin

Special Requirements: None (printer optional)
Files Used: FNPLOT.BAS

So you went out and bought a PC. You thought you'd be able to do all sorts of fancy displays and graphics. Then you discovered that you couldn't do these tricks without various exotic cards and special printers. Well, cheer up. Here is a program that will do some pretty fancy plotting without a graphics card, and will print these plots without a graphics printer. FUNCTION PLOT *does a lot of hard BASIC programming for you so that you can write a simple BASIC program to calculate mathematical functions.* FUNCTION PLOT *will then display the results in a graph.*

BACKGROUND

FUNCTION PLOT is written in BASIC. Part of this program allows you to add certain lines, which contain mathematical functions in

the form of Y=F(X). Once you have done this, you can run the original program to calculate the results of your function over a specified range, and to display the results as a graph, labelled to your taste. The plots may also be printed and tables made displaying specified values for up to four functions at the same time. Finally, you may save the specified functions on disk so that *FUNCTION PLOT* can call them up at a later session. Since this program uses only standard BASIC commands, it requires no special hardware or software.

START-UP

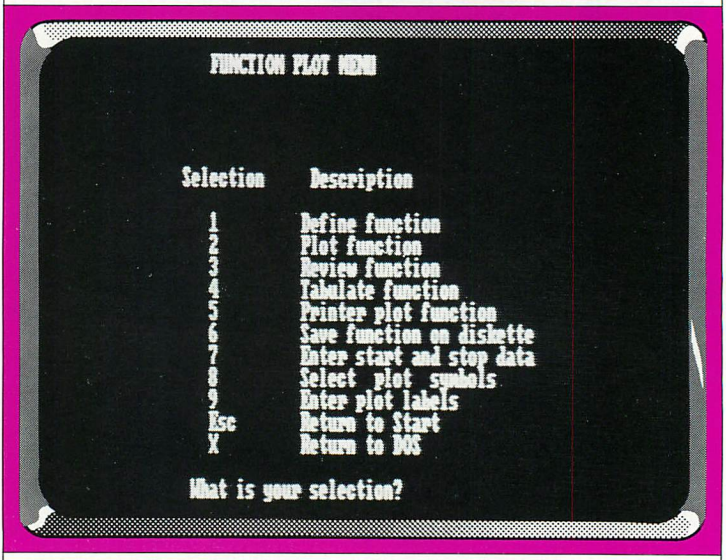
Put a system disk into your default drive and load Advanced BASIC by typing:

BASICA ↵

Then put your work copy of the *PC Disk Magazine* diskette in your default drive. Start the *FUNCTION PLOT* program by typing:

RUN "FNPLOT ↵

After a title screen, you will be shown the main menu.



The FUNCTION PLOT menu

TYPE THE FUNCTIONS AS BASIC COMMANDS

DEFINING YOUR FUNCTIONS

In order to define the functions you wish to plot, choose option 1 on the main menu. This choice will bring up a submenu with three options. Option 1 is used initially to type in your functions manually. Option 2, as we shall see later, allows you to call up and reuse a set of functions from a previous *FUNCTION PLOT session*. The third option tells you that you can return to the main menu by pressing the Escape key.

Option 1 on this submenu will bring you to the BASIC editing mode. You can now type in the section of the program that contains

your function calculation instructions. You should type in these functions as BASIC commands, starting on line 100 and finishing before line 950. The functions should be of the form:

```
100 Y1=function(X)
```

where the "function" is any combination of standard BASIC mathematical functions and operations. You can type up to four different functions on X, naming them Y1, Y2, Y3, and Y4. Once you have done this, press:

F2

Since you are in the BASIC editing mode, this key RUNS the current program, which means that *FUNCTION PLOT* is started up again. Now, however, the lines that you have just typed in are incorporated into the *FUNCTION PLOT* program.

PLOTTING THE FUNCTIONS

After you are returned to the main menu, choose option 2 (Plot function). You must first specify the labels and parameters for the entire plot, the Y-axis, and the X-axis, as well as the keyboard symbol you want to represent the points in each function. There are default symbols assigned for Y1 through Y4, but you may change these to any other letter, number, or symbol on the keyboard. Finally, set the range of values of X for which you wish to plot the values of Y. You will be asked for a start and stop value for X. Zero is the default start value, but you are required to enter an acceptable stop value.

Once you have set all of these parameters, the program will take a moment to perform the calculations on the functions. It will briefly display a message telling you that pressing the Escape key will return you to the main menu. Then it will display your plot.

THE FUNCTIONS ARE ALWAYS PLOTTED IN THE SAME SCREEN AREA

The functions are always plotted in the same size area of the screen, so the program will adjust the placement of the axes for readability. The scale on the Y-axis always goes from -1 to +1, but a minimum and maximum are given at the right side of the graph for each function in order to ease numerical interpretation.

The values along the X-axis are plotted in scientific notation if they are greater than ten or less than negative 10. This notation takes the form:

```
6X1.0E+02
```

where "6" is the number which is to be multiplied by a power of ten, "1.0E" signifies 10, and "+02" is the power to which ten is to be raised. So in the example, the number is 6 times 10 to the power of 2, or 600.

When the plot is displayed, press the Escape key to return to the main menu.

Let's try an example. To plot the function Y = 10X + 3, follow these steps:

(1) Load the program and run it. A menu will appear.

(2) Select choice 1. A choice of how to define your function is offered.

(3) Select choice 1. The screen clears and an "ok" appears.

(4) Type in the function in BASIC:

100 Y1 = 10*X + 3 ↵

(5) Type:

RUN ↵

or press:

F2

and the menu reappears.

(6) Select choice 2. Respond to each of the cues as follows:

Plot title:

LINEAR EQUATION ↵

Y-axis label:

AMPLITUDE ↵

X-axis label:

INDEPENDENT VARIABLE ↵

Accept the default plotting symbol by typing:

↵

X-axis start:

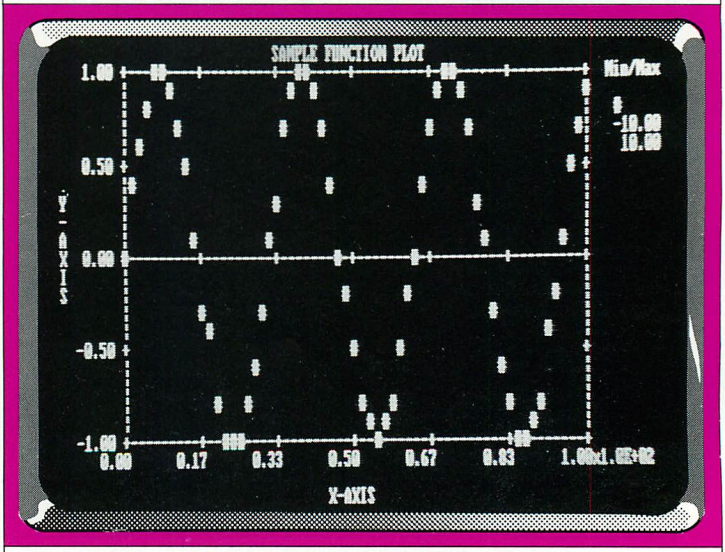
0 ↵

X-axis stop:

12 ↵

AFTER A SHORT DELAY, THE FUNCTION IS PLOTTED

There will be a short delay while the function is calculated and then the function will be plotted on the screen.



A plotted function

CHANGING THE PLOT

Your original parameters will reappear every time you plot a particular function. However, options 6, 7, and 8 on the main menu will allow you to change these parameters.

If you wish to change the functions themselves, press option 3 (Review function). This will display lines 100 through 995 of the program and put you back into the BASIC editing mode. You may now use the BASIC editing commands to change or add to your function lines. (See Chapter 2 of the IBM BASIC manual for details on BASIC program editing). When your editing is complete, run *FUNCTION PLOT* from the beginning again by pressing:

WHEN YOUR EDITING IS DONE, RUN THE PROGRAM AGAIN

SAVING FUNCTIONS

If you would like to save a set of functions for later use, choose option 6 on the main menu. You will be asked which drive contains the disk which will save the function. Enter the appropriate drive letter identifier. After you then name the storage file, *FUNCTION PLOT* will store the function lines of the program, using the extension ".FN" as part of the filename.

The next time you run *FUNCTION PLOT*, you can access the saved functions by choosing option 1 on the main menu (Define function) and option 2 (Read in functions) on the submenu. Then designate the drive where the storage disk is and type in the name of the storage file. (You need only type the name proper. The extension is not necessary.)

FUNCTION PLOT will read the lines into the program and return you to the main menu. The plotting parameters will have to be redefined, since they are not stored with the functions.

PRINTING THE PLOT

There are two ways to print a plot. If the plot is displayed, you can simply press:

You may also choose option 5 on the main menu, which will print the plot in a smaller type font and in a slightly condensed size. The information is the same in each format; you merely have a choice of sizes for your printed graph.

TABULATING THE FUNCTIONS

Option 4 on the main menu allows you to print a table containing values of each function for a series of values of X. First you must fill in parameters for the values of X, then specify a start and stop value and a size for the steps between each value. If you default on the size by pressing the Enter key, the program will pick a step size that will produce 60 entries between the start and stop values for X.

Once you have chosen the parameters, the program will print the table and prompt you to press the Escape key to return to the main menu.

THE ADVANCED FEATURE

FUNCTION PLOTTER can also include coefficients in your functions that can be entered anew each time you run the plotting program.

Lines 10 through 95 of the program are reserved for this purpose. Enter the BASIC editing mode by using option 1 (Define function) or option 3 (Review function) on the main menu. Write a coefficient input program in the reserved space. You *must* end this program with the following line:

```
95 GOTO 3000
```

When you define your functions in the function definition area of the program (lines 100 through 950), include these coefficients. Now, whenever you press F2 to run *FUNCTION PLOT*, your coefficient input program will run first. Input the desired numbers, and they will be included in any operations performed on the functions.

For example, if you want to plot the function y(x) = ax^2 + bx + c, where "a", "b", and "c" are to be input as new values each time you plot the function, you would write a program similar to the one listed below:

```
10 INPUT "ENTER A"; A$
20 INPUT "ENTER B"; B$
30 INPUT "ENTER C"; C$
40 a = VAL(A$): b = VAL(B$): c = VAL(C$)
```

Now each time you run the program, you will be prompted to enter new values for the coefficients "a", "b", and "c" in your function.

There is a chance that the *FUNCTION PLOT* program will crash if you use the advanced feature. This will happen if you have chosen a variable name for your redefinable coefficients that is also used by the *FUNCTION PLOT* program itself. You can avoid this by choosing variable names that begin or end with the letter "X".

If you ever wish to change the coefficient-input program while running *FUNCTION PLOT*, choose option 3 (Review function) on the main menu. This option will list only lines 100 through 950 of the program. But since you are in the BASIC editing mode, you can view the inputting program by using the BASIC command:

```
LIST 100-950
```

to see these lines and modify them.

YOU CAN EXIT THE PROGRAM IN TWO WAYS

If you use the "Save function on diskette" option when using the advanced feature, the advanced feature section of your program will also be saved on disk, and will run whenever you reload the functions from the storage disk.

EXITING

You can exit *FUNCTION PLOT* in two ways: by typing the BASIC command:

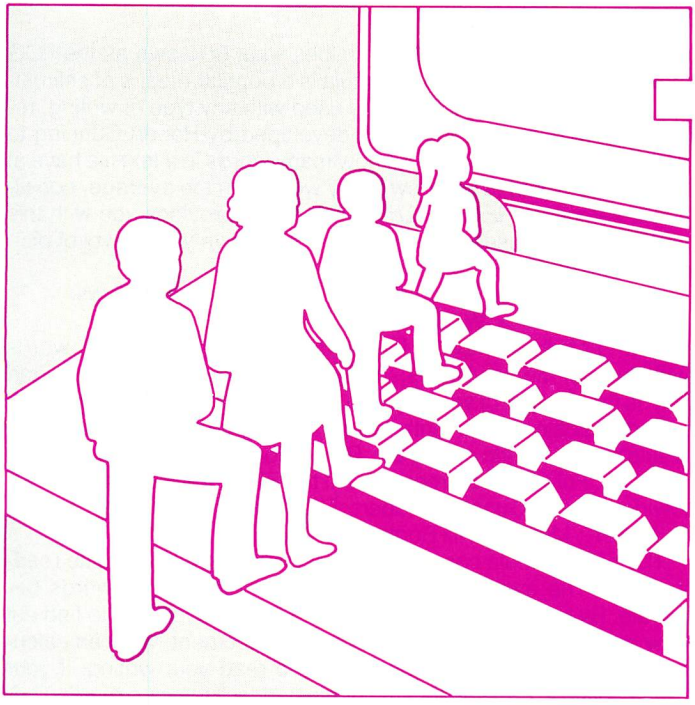
```
SYSTEM
```

when you are in the BASIC editing mode, or by pressing:

```
X
```

when you are on the program's main menu. Both of these options will return you to DOS. Caution—neither option will save any information from the *FUNCTION PLOT* session automatically. You must save functions explicitly during program execution.

PERSONAL



READABILITY ANALYZER

By Marty Franz and Michael Ezzo

Special Requirements: None
Files Used: FOG.EXE

Have you ever tried to explain something to someone, but find that he or she has little or no idea what you're trying to say? The generation gap is not the only deterrent to effective communication: maybe your word usage is simply too difficult for your intended audience to understand.

Readability is a measure of the complexity of a sample of writing. The readability level of a piece of writing influences whether that material will be read: if it is too difficult for the targeted audience, they may give up and stop reading. On the other hand, if it is too simplistic, the audience might also dismiss it as unimportant. One way to determine this difficulty level is to use a readability formula which calculates the reading level and expresses it as a grade level. You can use this readability formula to predict, quickly and fairly accurately, how well your targeted audience will understand what you have written.

Educators always try to order books that are challenging enough for their students without being discouragingly difficult. However, readability does not just concern librarians and teachers. Today readability has applications in educational systems, in mass and professional communications, and in the writing of insurance policies and government regulations.

BACKGROUND

READABILITY ANALYZER determines what is known as the FOG index of a text file. The FOG formula is a popular means of estimating readability because it may be used with any type of writing, regardless of its length. FOG was developed by Robert Gunning to derive a grade level based on how many words in a text file have at least three syllables and how many words, on the average, constitute a sentence. *READABILITY ANALYZER* provides you with this information and uses Gunning's formula to give you a level of difficulty immediately.

The formula used to compute the FOG index is as follows:

.4 (T/W * 100 + W/S)

where "T" is the number of long words in the document, i.e., words with three or more syllables; "W" is the total number of words; and "S" is the number of sentences in the document. A FOG index of 11 would indicate an eleventh grade reading level, an index of 12 would signify twelfth grade, etc. Rarely does the FOG index exceed 14 or 15. For more reference about the FOG index, consult the following source: Gunning, Robert, *The Technique of Clear Writing,* N.Y., N.Y.: McGraw-Hill Book Co., 1952.

THE FOG FORMULA CAN ALSO ESTIMATE READING TIME

The FOG formula can also use this information to estimate reading time. The average person reads at a rate of 200 words per minute. By knowing this rate and using the FOG formula to find out how many words are contained in your document, you can calculate how much time someone needs to read your writing. If your readers will be pressed for time (for instance, if they are reading advertising copy or technical instructions), you can alter your writing accordingly.

START-UP

To begin *READABILITY ANALYZER,* you must first insert a system diskette in your default drive and boot up DOS. Next, put your work copy of this issue of the *PC Disk Magazine* diskette into the default drive and type:

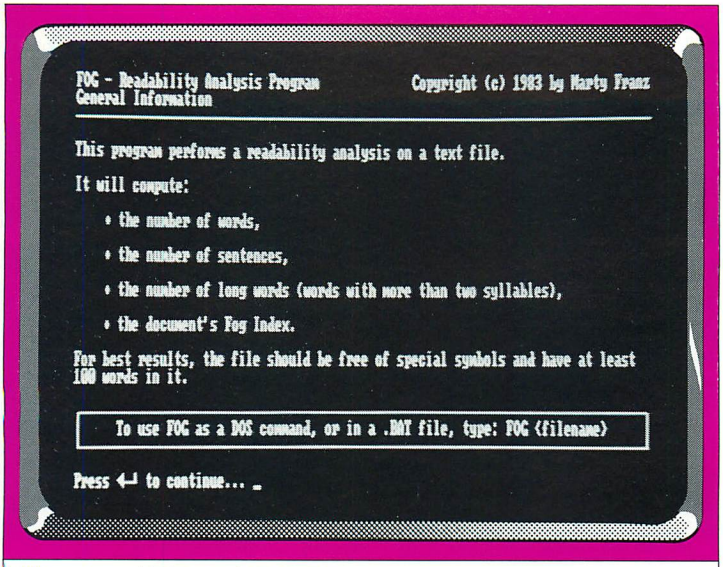
FOG ↵

in order to see the help screen describing what the program does.

The monitor first clears and then displays the main screen. As indicated by the message on your screen, the program will perform a readability analysis on a text file and compute the total number of words, the average number of words per sentence, and the total number of words of more than two syllables. It will also provide the text file's FOG index. The main screen will advise you that for best results, the file should contain at least 100 words and contain no special characters (i.e. no word processing files).

Those who don't need the help screen should type:

FOG filename ↵



The general information screen

where "filename" is the name of the file you want analyzed. Typing the above command speeds analysis by bypassing the help screen and immediately printing the results on the screen.

ANALYSIS

If you do not specify the name of a file to analyze when you bring up the program, proceed by pressing:

after you've read the first screen. The program instructs you to "Please type text file name (just Enter to quit)." Simply type the name of the file you want analyzed, and *READABILITY ANALYZER* goes to work.

If you make any mistakes while typing the filename, the program will be unable to find your file and will prompt you to enter the name again. If you choose, you can also exit the program at this point by pressing:

without typing in a filename. This action puts you right back into DOS and you can then start over.

Since the program takes several seconds to perform an analysis, it will reassure you that all is going well with the "Analyzing file <filename>" message.

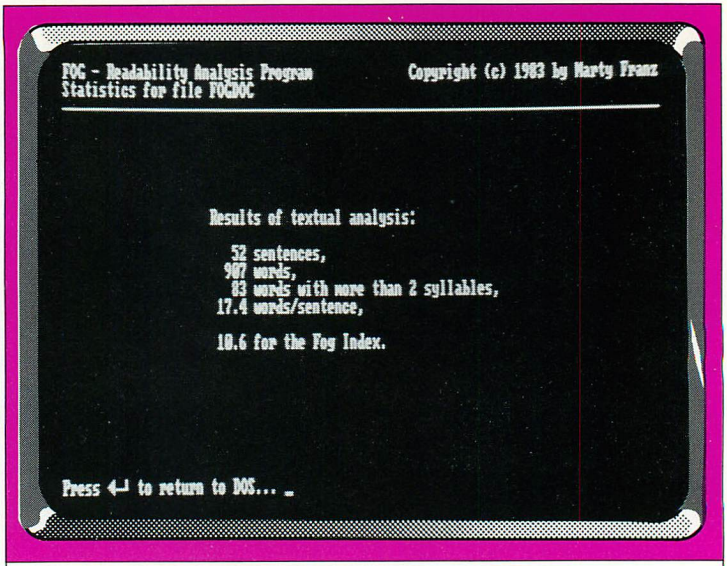
YOU MAY SEND YOUR RESULTS DIRECTLY TO THE PRINTER

PRINTOUT

READABILITY ANALYZER includes an option that lets you send your results directly to the printer instead of the screen. After the "A>" prompt, type:

```
FOG filename >PRN:
```

where "filename" is the name of the file you want analyzed. This command will produce a hard copy of your readability analysis.



Results of a textual analysis

STORING

A final procedure stores the results of the analysis in a file without displaying them on the screen. After the "A>" prompt, type:

FOG filename1 > filename2

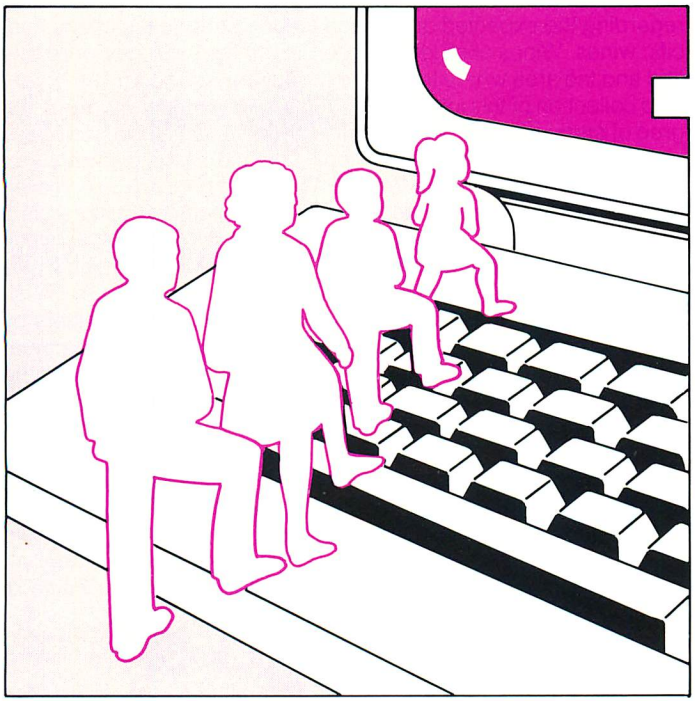
where "filename1" is the name of the text file to be analyzed, and "filename2" is the file which will store the results. One word of caution—this option erases any previous data stored in "filename2" so you must be careful when entering a filename in which to store your results.

USE CAUTION WHEN STORING YOUR RESULTS

RESULTS SCREEN

The final screen shows you the "Results of Textual Analysis." The program first lists the total number of sentences. Beneath this appears the total number of words, then words with more than two syllables, the average number of words per sentence and, finally, the FOG index. When you are finished, just press:

to return to DOS.



WINE DATA 2

By Morris Effron
Programmed by Henry Bilenchi

Special Requirements: None
Files Used: WINE2.DAT
WINE2.BAS

Do you know a Barolo from a Barbaresco? After you've spent ten minutes studying a restaurant wine list, do you still lack the courage to order anything but a glass of water? And when your host talks about "a wonderful bouquet," do you think he is describing the centerpiece? Well, your days of wine and neurosis are over. PC Disk Magazine *brings you the second installment of* WINE DATA, *a series of data files containing a wealth of information about the wines of various countries. This issue treats the wines of Italy and promises you enough oenological knowledge and confidence to impress your friends, stock your cellar, and terrify wine stewards.*

BACKGROUND

Hundreds of unpredictable factors determine the quality of a wine —the weather, the composition of the soil, the time of the grape harvest, the bottling techniques and materials, and the wine's age. These variables make today's selection of wines richly varied in quality and cost, and often hopelessly confusing to the uninitiated.

Fortunately, there is some agreement among wine connoisseurs regarding the expected quality and the required aging time of specific wines. Wines are judged according to the year they were bottled and the area where they were produced. Armed with an organized collection of this information, you can determine with some degree of certainty whether any particular bottle contains a good wine or not. You could also tell whether it should be enjoyed now, or stored for further aging. In *WINE DATA 2,* the consensus on Italian wines has been consolidated into a rating system and stored in a data file. In contrast to the French wines of *WINE DATA 1,* which were examined by year and region, the Italian wines studied in this program are not easily categorized. Some of the wines are classified by the type of grape, some by family name, and still others by growing region. For our purposes, we will list the Italian wines by label and by year (or vintage), thereby allowing you to access the wine data file by either year or label. You may also use the Print Screen capability of the IBM PC to make a hardcopy of any of the rating displays.

FIND OUT IF IT IS A GOOD WINE OR NOT

You will notice that only red wines are listed in this program. This is because the tremendous variety of Italian red wine far overshadows that of Italian white wines. In general, the Italian whites should be drunk when they are as young and fresh as possible. Three Italian whites, however, improve with a few years of age: Fiano di Avellino, Greco di Tufo, and Vernaccia di San Gimignano.

START-UP

Load Advanced BASIC into your PC by putting a system disk into your default drive and typing:

BASICA ↵

Then put your work copy of the *PC Disk Magazine* diskette into your default drive and type:

RUN "WINE2 ↵

to display the main menu of *WINE DATA 2*. This menu has three options. The first option rates the wines of a particular vintage (year) by wine label. The second option rates the wines of a single label by year. Option 3 lets you exit the *WINE DATA 2* program and return to Advanced BASIC.

RATINGS BY YEAR

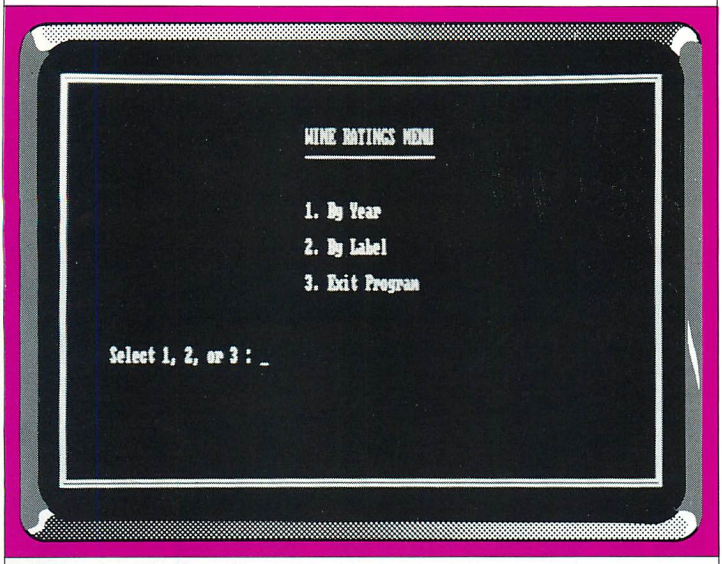
If you choose option 1 on the main menu, a prompt on the bottom of the menu screen will ask you which year you wish to see rated. Type in a year from 1961 to 1980 and press:

↵

A screen will now list the various wine labels of Italy. Next to the column of regions will be a column of ratings followed by a second column of one-letter codes. The number is the rating for the wine for

that particular label for that particular year. The letters are a code to give further information about the ratings. At the bottom of the screen is a legend explaining the ratings and codes.

The ratings range from 1 to 10. Ten is the best rating a wine can receive, and 1 is the worst. Wines rated 5 can be considered aver-



Choose ratings by year or label

THE WINE RATINGS RANGE FROM 1 TO 10

age, wines rated below 4 are poor, and those above 7 are quite good.

The letter codes concern the timeliness of that particular wine. Wines that have had just the right amount of aging should be enjoyed now and are therefore labeled with an "N." Certain wines which may benefit from further aging will carry the additional label "A." Wines that are still too young to drink will have a "Y" after the rating; conversely, wines that have reached old age are labelled "O." Naturally, these labels are based on the subjective judgements of experts, just as the ratings themselves are, and should not be taken as absolutes.

There is one additional database in *WINE DATA 2*. This contains additional information on each wine label. This data can be accessed from either the Ratings by Label or Ratings by Year display. From the Ratings by Year display, you must first select the label for which you would like this additional information. Notice that the first label on the list is highlighted. The highlighted label is the one currently selected for additional information. You can change the label selected by pressing:

8 ↑ or 2 ↓

to move the selection highlight up or down respectively. When the label you want is highlighted, just press:

F1

to view the background information on that label. Then, press:

Esc

to return to the Ratings by Year screen. To exit from the Ratings by Year screen and return to the main menu, just press:

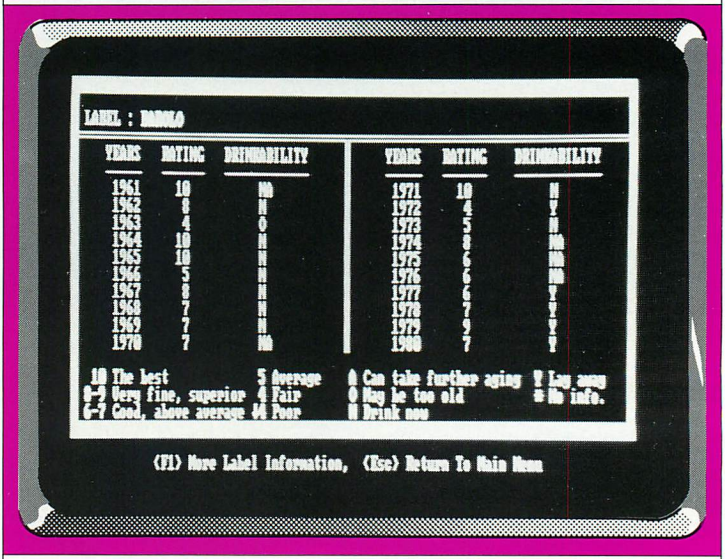
Esc

when at this screen.

RATINGS BY LABEL

LIST ALL THE LABELS COVERED IN THIS DATA FILE

To see the wines of a particular label rated by year, choose option 2 on the main menu. A prompt will appear at the bottom of the main screen along with a numbered list of the labels covered in this data file. Answer the prompt by entering the number of the label you wish



Ratings by label

to examine (there is no need to press the Enter key). The screen will then display the wine ratings for that label for all years covered by the data file. The rating codes used here are the same as those in the Ratings by Year display.

From this screen it is also easy to access further information on a label. Press:

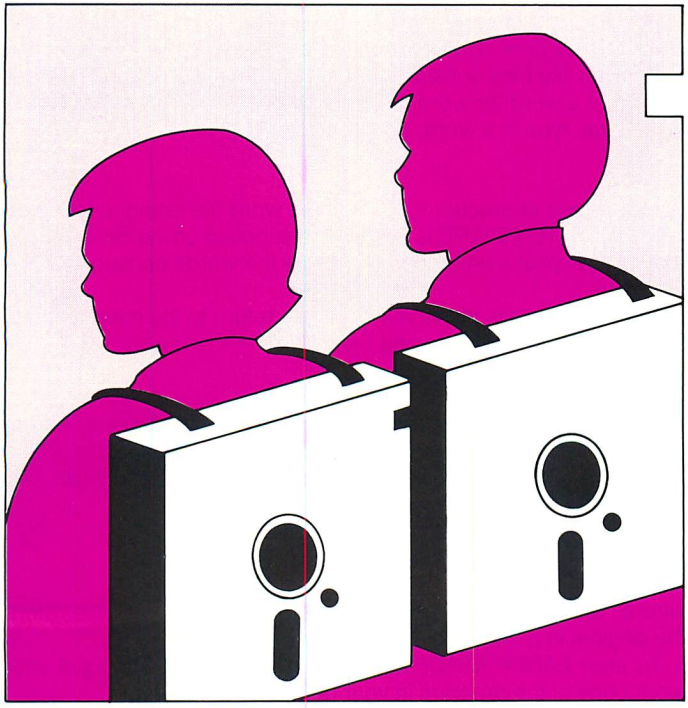
F1

and on a new screen will appear background information on the wine with that label. Pressing:

Esc

returns you to the Ratings by Label display. Press the Escape key again to return to the main menu. You may then exit the program from the main menu by choosing option 3.

EDUCATION



EARLY READER

By Bo Reahard and Pat Gleason

Special Requirements: Color/Graphics Adapter, 128K RAM
Files Used: ER.EXE
ER.DAT
TOYS.WRD

Even though most children want to learn to read, their attention often wanders from the tiny, black and white print of traditionally formatted books. EARLY READER *is a program that uses large, colorful words to make reading engrossing, thereby teaching children to read before they start school and familiarizing them with the academic trend of the future, the use of a personal computer.*

It's easy and it's fun. EARLY READER *will help preschoolers recognize letters, sounds, and whole families of words after only a few hours of use. The program also allows the parent or teacher to use* EDLIN *or any other text editor to create additional word sets. Repeated exposure to these files enhances vocabulary skills.* EARLY

READER *offers an opportunity to spice up spelling drills that were once only accomplished with dull black and white flashcards.*

PROGRAM STRUCTURE

All the functions of *EARLY READER* branch off the main menu, which will appear after the program loads the character set.

At the main menu there are two initial options you may select. These choices change the background color and/or change the color of the letters in the screen display. Once you have chosen the colors you like best or have accepted the default colors, you can either enter a word (or words) of your choice or display a list of word files. If you type in a word, press:

to display the character. If you call the word file listing, choose an existing file (TOYS.WRD is the only file supplied on the disk, but you may create your own) and then display the words contained in the word file.

Once the word is displayed, you may return to the main menu to enter a new word by pressing Enter.

START-UP

BE SURE YOU ARE RUNNING ON THE COLOR MONITOR

To run *EARLY READER* from its own diskette, first transfer it from the *PC Disk Magazine* diskette to a blank formatted diskette. Place the *PC Disk Magazine* diskette into the default drive and type:

COPY ER.EXE B: ↵

COPY TOYS.WRD B: ↵

where "B:" is the drive holding the formatted work diskette. Store the original in a secure place.

To start *EARLY READER,* you must be in DOS. Then put your work copy of the program in your default drive and type:

ER ↵

Be sure you are running on the color monitor. The program will display the characters as they are loaded. The program then displays the main screen.

THE MAIN SCREEN

The main screen will accept any typed string of alphabetic characters to display. To view a word, type it onto the main screen and then press the Enter key. The display screen will appear and the word will be spelled out in large letters. To return to the main screen, press the Enter key. Repeat this procedure to view other words.

Note that *EARLY READER* uses as much of the screen as possible. Longer words print out with small letters and short words use large letters. The length of a character string is limited to 20 to 30, depending upon the size of the characters. In addition, after a string reaches 11 characters in length, the letters will only be outlined, not filled in.

Changing display colors is simple and enjoyable for the young reader. To change the color of the letters, press:

Ctrl L

When the cursor moves to the top of the screen, you may select any of the colors displayed. Type in the number and press Enter. To select the background color, press:

Ctrl B

and select any color displayed.

WORD FILES

When *EARLY READER* is loaded, no word files are resident in the program. Your *PC Disk Magazine* comes with one such file, TOYS.WRD, which contains a list of toy names. *EARLY READER* provides a word file submenu which you can call by pressing:

Ctrl F

The cursor will be positioned on the TOYS.WRD file. To select it, press the Enter key. This will return you to the main screen. Enter a number on the entry line and the program will draw that word in that file on the display screen. Selecting words by number eliminates spelling problems and minimizes the number of keystrokes needed to "flashcard" the word onto the screen. To advance to the next word in the file, press:

+

Similarly, press:

—

to go back a word.

You can create new files with the *EDLIN* line editor that accompanies PC DOS. Use the TOYS.WRD file as a model and be sure that all such word files bear the file extension ".WRD". You can print out the TOYS.WRD file with the DOS level command:

```
COPY TOYS.WRD PRN:
```

EXITING

Simply press the Escape key at any point to end the program and return to DOS.

F4 ALL THE WORD FILE SUBMENU

PROBLEM HANDLING

We try our best to thoroughly test all *PC Disk Magazine* software, and provide instructions that cover all aspects of its use. Nevertheless, error-free software and exhaustive documentation are elusive goals. So if you have a problem, please contact us and let us help. Although we hope you will not need it, the address to write to is:

PC Disk Magazine
Problem Recovery
One Park Avenue
New York, NY 10016

MERGING THE CODE

To move a block of code from one BASIC program to another, you must first move the code to be inserted into an ASCII file. Then load the destination program into BASIC, and lastly MERGE the ASCII file into that program.

The procedure seems simple. The twist, however, is that if there are any matching line numbers in the ASCII file and the current BASIC program, these lines of the current BASIC file will be replaced when you MERGE. Therefore, before you create your ASCII file, you should first eliminate all the extraneous lines in the program with the DELETE command, so only the section of interest to you remains in BASIC. Then use the RENUMBER command to set new lines for this extract, line numbers which you know will not conflict with those in the destination program. After renumbering, do your SAVE to create the ASCII extract file. Then load your program into BASIC and MERGE the extract file in.

An example using the *DIRECTORY READER* in this issue may be helpful. When in BASIC, put the *PC Disk Magazine* diskette in the default drive and type:

LOAD DIRREAD ↵

As discussed in the article on *DIRECTORY READER,* the two routines of interest are in lines 100-220 and 1000-1050. So the next step is to delete all extraneous lines. Type:

DELETE 300-330 ↵

DELETE 2000-9010 ↵

With the deletion done, renumber the extract with a starting line number well above any lines in the destination program. Type:

RENUM 10000 ↵

Then save the extract to an ASCII file by typing:

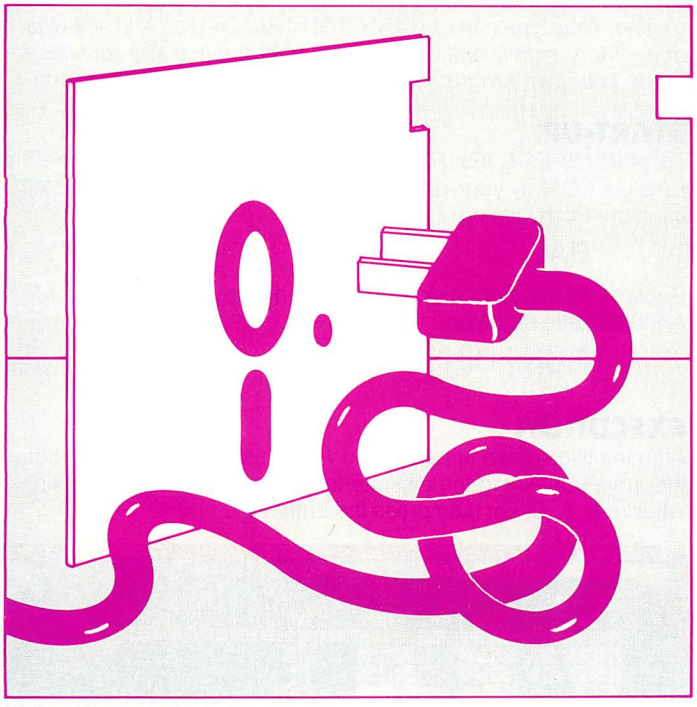
SAVE *"somename"*,A ↵

Lastly, load your program into BASIC, and "merge in" the extract file. Type:

LOAD *"yourfile* ↵

MERGE *"somename"* ↵

UTILITIES



DIRECTORY READER

By C-Level Software

Special Requirements: None
Files Used: DIRREAD.BAS
DIRREAD.BIN

DIRECTORY READER *is part of a continuing series of BASIC utility programs offered through* PC Disk Magazine. *These programs are designed to simplify working and programming in the BASIC language. In future issues you can look forward to subroutine sort programs as well as a library of error traps.*

The present program is a BASIC utility that reads the directory of a diskette or hard disk into a series of BASIC variables. This makes the directory information immediately available to your BASIC program for manipulation, unlike the BASIC "FILES" command which only displays such information. If you do any programming of your own, DIRECTORY READER *is a must for any file management programs you might create.*

DIRECTORY READER *consists of an assembly language subroutine, DIRREAD.BIN, which actually retrieves the directory information, and a BASIC program, DIRREAD.BAS, which calls this subroutine. The BASIC program contains two key sections, variable initialization and directory retrieval, which will be detailed further on. To demonstrate this utility, we have added a routine that allows you to specify the drive and file(s) you want retrieved, along with a display routine. Thus, you can run* DIRECTORY READER *as is to see what it does. More significantly, you can extract the two key routines for use in your own program and thereby utilize this file information.*

START-UP

To start *DIRECTORY READER,* put a system disk with the file BASICA.COM in your default drive. Then load Advanced BASIC into your PC by typing:

BASICA ↵

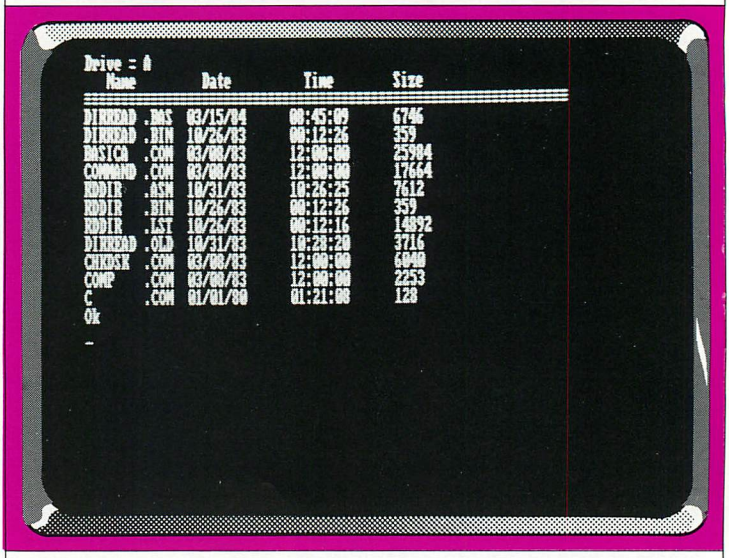
Replace your system disk with your work copy of the *PC Disk Magazine* diskette and type:

RUN "DIRREAD ↵

EXECUTION

After the title screen appears, you will be asked to enter the letter of the drive containing the diskette to be displayed. Type in a single letter (e.g. A, B, C), then press the Enter key.

FIND DATA FOR ONE OR MORE FILES



A complete disk directory

You are then asked to enter a file description for the file(s) you want displayed. With *DIRECTORY READER,* you can retrieve information

for a single file, a group of files, or all files in the designated drive. To display information for all files, simply press:

in response to this prompt. To retrieve a single file's information, enter the full filename (with its extension) followed by the Enter key. To retrieve a particular group of files, you can use the wildcard characters "?" and "*" provided by DOS. For example, to display a list of all the BASIC program files, you could reply:

*.BAS 

To display a list of all files that begin with the letters "TESTP", you could reply:

TESTP???.* 

This facility lets you selectively retrieve information for files of a particular type.

Once you have entered a file description, the program proceeds to retrieve the relevant information and then display it on your screen. The information displayed, "Name," "Date (of last update)," "Time (of last update)," and "Size (in bytes)," is the information available in the disk directory.

If the number of files to be displayed exceeds one full screen, entries will begin to scroll out of view. You can halt the scrolling feature at any point by pressing:

Resume the display by pressing any key.

When the display finishes, the program ends execution and you will see the "OK" prompt indicating that you are back in Advanced BASIC. You can run the program again by simply typing:

RUN

Or you can type:

SYSTEM 

to exit Advanced BASIC and return to DOS.

USE THE WILDCARD SYMBOLS "?" AND "*" PROVIDED BY DOS

THE PROGRAM

To retrieve directory information in a program of your own, you need three key elements from *DIRECTORY READER:*

1. The assembly language subroutine, DIRREAD.BIN.
2. The variable initialization routine from DIRREAD.BAS.
3. The retrieval routine from DIRREAD.BAS.

Getting the assembly language subroutine is easy. Just copy the file DIRREAD.BIN to the diskette where your program resides.

The variable initialization routine of DIRREAD.BAS is contained in lines 100-220. Several string variables are dimensioned to accumulate the file information retrieved (each call to DIRREAD.BIN retrieves the information for one file). FLNAME$ accumulates the full filenames retrieved singly with each call. DATE$ accumulates the date of last update information, TIME$ the time of last update, and SIZE$ the file size. A counter variable, FLCOUNT%, is used as a subscript for these variables during retrieval. After retrieval, it con-

tains the total number of files retrieved. These variables are dimensioned at 112, which is the maximum number of files any single directory on a double-sided diskette can hold. There is no limit to the number of files a hard disk directory can hold (it's limited only by available disk space), so you may want to increase the dimensions of these variables when using DIRECTORY READER with a hard disk. Then a collection of individual variables are initialized for the first call to DIRREAD.BIN. The two key variables that you will want to set are A$ and DRIVE%.

EACH QUESTION MARK MATCHES A CHARACTER

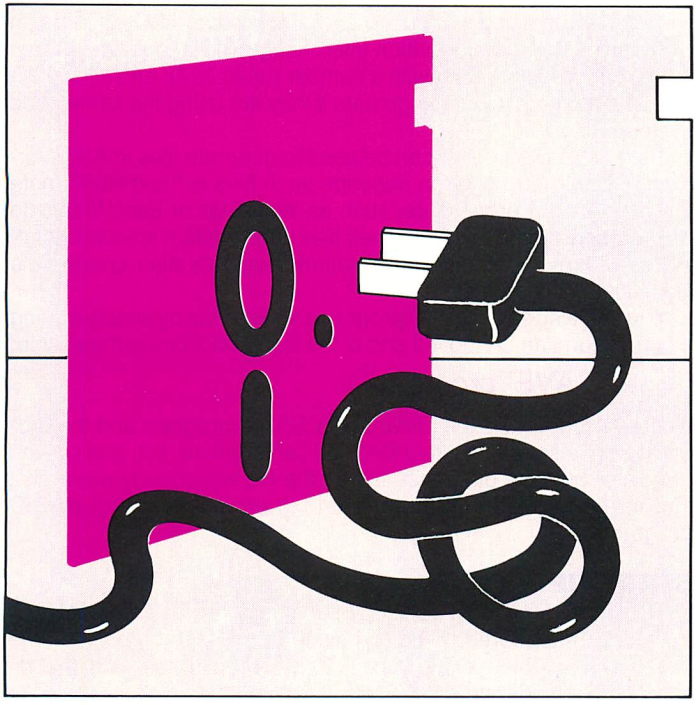
A$ contains the file description, or retrieval "screen" if you will. It is an 11 character string, allowing an eight character filename and a three character extension (the intervening period is not used). Putting a character anywhere in this string requires that a filename have the same character in the same position to be retrieved. A question mark matches any character (thus a string of 11 question marks retrieves all files).

DRIVE% contains a number indicating which drive is to be accessed for directory information. 1 = A, 2 = B, etc.

The information retrieval routine is contained in lines 1000-1050 of DIRREAD.BAS. The crux of this routine is the call to DIRREAD.BIN. As stated above, each call returns one file's worth of information. After the call, the variable STAT% tells us if information was retrieved (STAT% = 1), or if we have already read the last file entry (STAT% = 0). In the latter case, we exit the retrieval routine with a GOTO (you'll probably want to change the line number branched to in your own program). If we have retrieved information for another file, it is put in the accumulating array variables, and our file count is bumped. The variable INIT% serves as a flag indicating whether this is the first call to DIRREAD.BIN or not. The flag is turned off (= 0) after the first call. Upon exiting this retrieval routine, our directory information is contained in the variables FLNAME$, DATE$, TIME$, and SIZE$, and the number of files retrieved is in FLCOUNT%.

To incorporate these two routines from DIRREAD.BAS in your own program, we refer you to the article "Merging the Code" in this issue.

UTILITIES



FILE MANAGER II

By C-Level Software

Special Requirements: *FILE MANAGER I,* Printer
Files Used: FM.COM (from last issue—Volume 1, Number 5)
FM.EXE (from last issue—Volume 1, Number 5)
FM.HLP (from last issue—Volume 1, Number 5)
FM.LOG (from last issue—Volume 1, Number 5)
FMPRNT.LOD
FMPRNT.HLP

"Just what do I have on this disk?"

Last month, PC Disk Magazine *introduced a utility designed to help answer this question.* FILE MANAGER I *let you first list all the files on a disk, and then scroll through the list to find specific files. The program also made it simple to rename, copy, or delete a file in the directory.*

But another problem remains: how do you get a hard copy of the text files so that you can refer to them later, either for reference or

revision? This month's PC Disk Magazine *includes the second part of the* FILE MANAGER *series, which allows you to execute the commands of Part I (rename, copy, and delete), as well as print a listing of the text file that you select.*

BACKGROUND

The print features of *FILE MANAGER II* work with text files only. Text files store data using a standard format known as ASCII (American Standard Code for Information Interchange). This format identifies each keyboard character with a number (up to 127), and allows different computers to exchange data if they are using the same code system.

The EDLIN DOS editor can be used to generate files in ASCII format. Another term used to describe such files is "text files" (note that some word processors, such as WordStar or EasyWriter, do not normally create text files. Their files often contain special control codes). Many database management programs also create data files in a text file format.

SAVE BASIC PROGRAMS AS TEXT FILES

You can also save BASIC programs as text files by simply adding a comma and an "A" to the end of the filename. For example, type:

SAVE"program",A ↵

where "program" is the name of our BASIC program and the program will be saved as a text file. You can still load, list, and run the program; the only difference is that the file will occupy more disk space than it would in the normal tokenized format of BASIC programs.

START-UP

FILE MANAGER II must be used in conjunction with *FILE MANAGER I,* which appeared in the last issue of *PC Disk Magazine*. The best way to do this is to copy the *FILE MANAGER* files from both issues onto a separate disk. Place this new disk into your current drive. If you just want to use *FILE MANAGER I,* type:

FM ↵

To invoke *FILE MANAGER II,* which will now include all the features of *FILE MANAGER I* plus its own print features, type:

FM P ↵

FILE MANAGER II will start by displaying the title screen. At the bottom of the screen you will be prompted to "Press ESC to continue." Doing so will take you to the main screen.

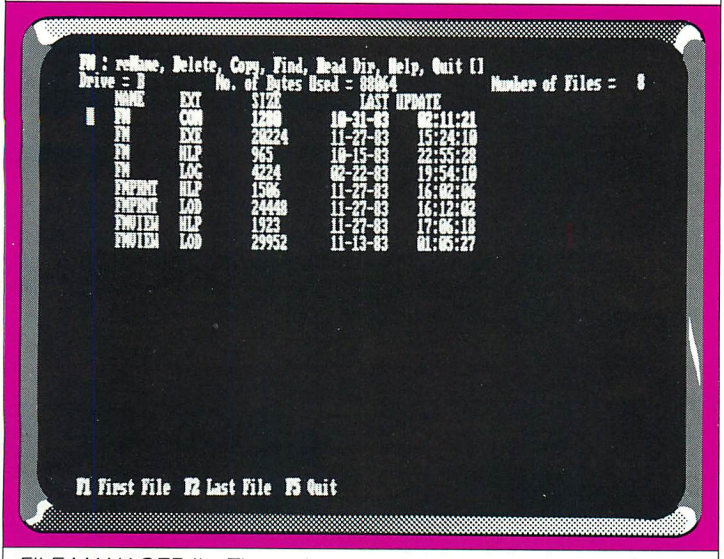
THE MAIN SCREEN

This is the standard *FILE MANAGER* main screen, which contains three sections: the Command Line, the Display Window, and the Function Key Line.

The Command Line shows the list of available commands: "reName", "Delete", "Copy", "List", "Find", and "Help". Select these by typing the corresponding capital letter for each. You will notice that after the Help command, there is an ellipsis (three dots). This means that there are more options available. Press the Enter

key to display on the Command Line the other available commands: "Read Dir", "Map Dir", "Abort Print", "Print Params", and "Quit". Then press the Enter key once more to read the first six options again.

The Display Window shows the list of files on the disk. On the second line, right below the Command Line, the screen will show the letter of the current drive, the number of bytes used on the disk, and the number of files. The next line displays the headings for the list of



FILE MANAGER II—The main menu

files: NAME, EXTension, SIZE, and the date and time of the LAST UPDATE. The screen will alphabetically display up to 22 files at one time.

The Function Key Line shows which function keys are active, and what they will do.

THE NEW COMMANDS

The commands "reName", "Delete", "Find", "read Dir", "Help", and "Quit" all work the same as in *FILE MANAGER I*. (See the manual for *PC Disk Magazine,* Volume 1, Number 5 for an explanation of these commands.) The new commands in *FILE MANAGER II* are "List", "Map Dir", "Abort Print", and "Print Params". These allow you to print a copy of your directory and text files on your printer.

The "List" command prints a selected text file on the printer. You start by using the *FILE MANAGER I* commands to select a file from the directory listing. Then type:

L

and the program will ask if you want to have "Headers" or "No headers" or if you want to Quit. If you quit, you will abort the "List" command and return to the original list of command options. If you select the "Headers" option, the program will print a listing of the file on your printer with the filename, the page number, and the cur-

THE LIST COMMAND PRINTS TEXT FILES

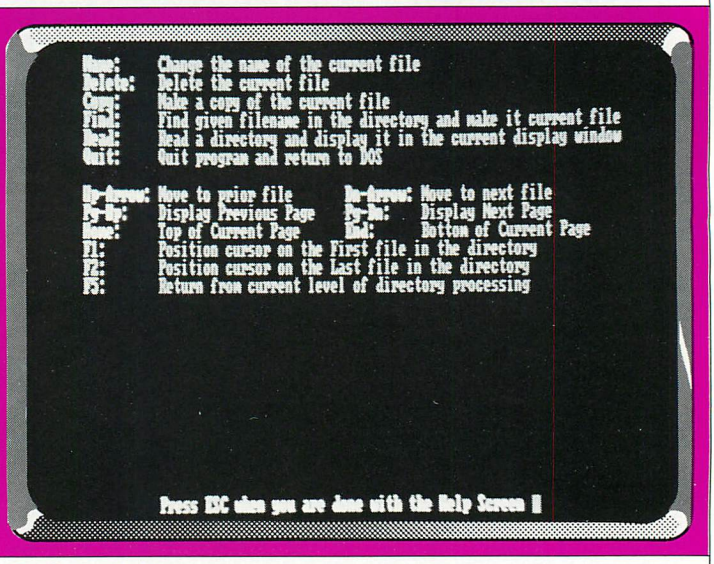
file on your printer with the filename, the page number, and the current date and time at the top of each page. These may be omitted by selecting "No headers" as the List option.

THE ABORT PRINT COMMAND STOPS A PRINT IN PROGRESS

One important feature of the "List" command is that printing occurs concurrently with other *FILE MANAGER* operations. In other words, as soon as you finish with the "List" command, the program will return to the main command options, and will print out the file while you continue to use the program. This feature of *FILE MANAGER* was designed to allow you to use your time more efficiently because it spares you from the frustration of idly watching while a long printout is being performed.

The "Map Dir" command is similar to the "List" command, except that it prints a listing of the directory. The listing includes the drive letter, the number of bytes used, and the number of files, just as they are displayed on the main screen. Each page also shows the time and date of the listing.

You may stop a printing operation using the "Abort Print" command. The program will show which file is currently being listed on the printer, and let you abort either the current file or all the files scheduled to be printed. You may also ask for Help, or Quit the Abort command, which resumes the listing process.



The help screen

The "Print Params" command allows you to set the print format parameters for the printed listings. You may specify the line width or the page length (or you may Quit). The default line width is 80 characters, which is eight inches long when using 10 characters to the inch. If you select the line width option, you may choose any positive integer value. Similarly, the default page length is 62 lines per page, but you may change this by choosing the page length option. Finally, you may select Quit to return to the main command options. You may quit the program itself by using the main screen command "Quit".

GAMES

CHAMELEON

By Casey Roche

Special Requirements: None
Files Used: CHAM.BAS
BEST

After all those video games of shooting aliens in UFOs and dodging bombs, isn't it time for a change? Instead of looking into the future, CHAMELEON *is an adaptation of the ancient board game, "Go." The idea is to outflank your opponent's colored pieces and to flip them so that they change to your color. The trick is to claim the best squares for yourself, and block your opponent, either the computer or another person, from doing the same.*

START-UP

To start *CHAMELEON,* you must first load Advanced BASIC by typing:

BASICA ↵

Then put your work copy of the *PC Disk Magazine* diskette into the default drive and type:

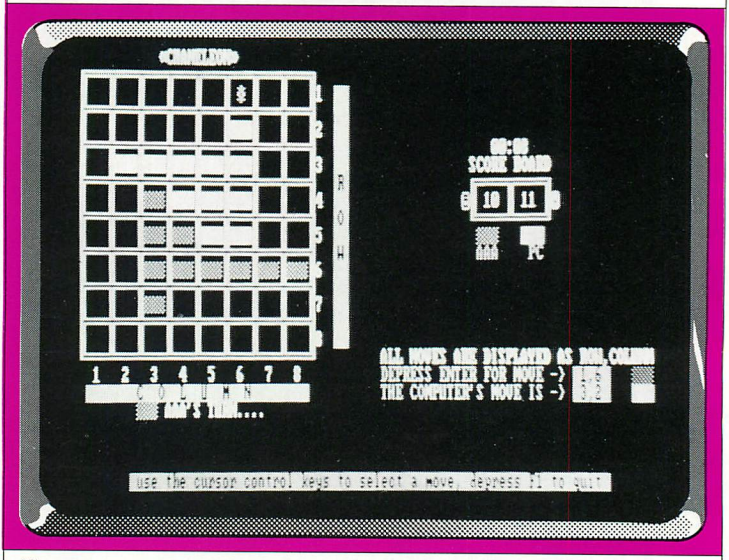
```
RUN "CHAM ↵
```

PLAY AGAINST ANOTHER PERSON OR AGAINST THE COMPUTER

After the title page appears, you will be asked if you want sound effects. Then choose whether you want to play against another person or against the computer. If you choose to play the computer, you must also select a difficulty level: beginner, intermediate, or advanced. (Play the advanced version at your own risk!) Finally, the computer will flip a coin to see who goes first.

THE GAME SCREEN

The game screen is dominated by a large checkerboard-like grid of eight rows and eight columns. To the right of the game board are a



The game screen

clock, a scoreboard, and a status display. The clock is merely a convenience—there are no time limits in *CHAMELEON*.

Players are represented by solid or lightly shaded "pieces" that fill the squares on the grid. When the game begins, the four center squares are already filled in with two squares for each player. If you are playing the computer at the beginner level, the upper left and lower right squares will also be filled in with your pieces as a bonus.

PLAYING THE GAME

The squares in *CHAMELEON* are described using a pair of numbers that indicate row and column, in that order. A blinking cursor marks the active square. As you move the cursor around using the arrow keys on the numeric keypad, the position of the active square will be noted in the status display for the player whose move is due.

The goal of *CHAMELEON* is to surround your opponent's pieces and flip them so that they change to your color. You must position the cursor so that one or more of your opponent's pieces lies in a

straight horizontal, vertical, or diagonal line between the currently active square and another one of your pieces. In order to register your move, press:

Your opponent's surrounded pieces will change to your color, and the active square will be filled with one of your pieces. The scoreboard will reflect the new balance of territory.

If you try to enter an occupied square as your move, the program will reject your choice. And if you try to make a move which doesn't flip at least one of your opponent's pieces, the computer will reject it by saying "Illegal move—you must flip something." Examine the board again, and try another square. If the computer prompts you for a move, then it must be possible to make one.

After every move, the computer analyzes the board to make sure a move is possible. In some cases, one player may be unable to move, while possible moves may still exist for the opposing player. The program will skip the first player's turn and prompt the second player (if you are playing a fellow human), or automatically make its own move (if you are playing the computer). If no move is available, the game will end. The game will continue until all the squares are filled, and the player with the most squares wins.

At the beginner and intermediate levels, the computer doesn't examine strategy very carefully. It merely chooses the move that flips the maximum number of your pieces. If there are two or more moves that satisfy this requirement, the computer chooses randomly among them. (The only difference between beginner and intermediate is the two bonus squares you are given as a beginner.) At the advanced level, the computer evaluates all its possible moves and bases its selection on strategy. Should you be lucky enough to defeat the computer, you will be allowed to enter your name and the date and they will be saved for posterity.

USING STRATEGY

The object of *CHAMELEON* is not only to get as many of your squares on the board as possible, but also to limit your opponent's choices. Try to anticipate your opponent's moves so that you can prevent him or her from occupying valuable squares.

But hold your fire at first! If you have too many of your pieces at the center, you leave yourself vulnerable to massive flips as your opponent outflanks you. So flip one piece at a time at first, and build toward corners and boundary squares. This way you limit your opponent's ability to outflank you, and you maximize the number of directions from which you can attack your opponent.

ONE GOOD MOVE CAN REVERSE YOUR FORTUNES

EXITING

After every game, the computer will ask you if you want to play *CHAMELEON* again or quit. To quit at any time during the game, press:

to exit to Advanced BASIC. You can then either load and run another BASIC program, or return to DOS by typing:

```
SYSTEM
```

ROBLEM RECOVERY

There really is not a great deal to say about problem recovery with *PC Disk Magazine*. If you use this software on the right equipment running the appropriate system software, as outlined in the Technical Preface, you should experience no problems. Nevertheless, a few comments may resolve some more obvious difficulties.

Any BASIC program can be interrupted at any time by pressing:

Ctrl Scroll Lock

If you do not see the "Ok" message immediately, indicating that you are back in BASIC, press these keys again. This is a rather drastic but effective way of regaining control of the computer. You won't damage any of the programs in this way, since they're still intact on the diskette. However, you will lose any data you entered while the program was running.

If you interrupt a program you may find that the function keys no longer perform as they had before starting the program. This is because many *PC Disk Magazine* programs reset the function keys during execution, then restore the original settings upon completion. An interrupt causes an abnormal termination of a program, so the function keys are not restored. To correct this situation, simply exit from BASIC and then return to BASIC.

You may find at times that the cursor control keys are not working as they should. This is because the keys are not in cursor control mode. The key that switches these keys between numeric mode and cursor control mode is the Num Lock Key. So to restore the keys to cursor control mode, press:

Num Lock

If you try to send something to the printer when there is no printer, or when the printer is off or offline, you will hang the system. The computer will just sit there and will not respond to any keys pressed. After a few seconds you may get a BASIC error message indicating that the device was unavailable. The program that was running has been aborted, and you will be left in BASIC. If the computer does not put out any message, but just remains hung, you will have to say good-bye to whatever you were doing and re-boot your system.

Though we hope you will never need it, if you should find a "bug" in a *PC Disk Magazine* program, the address to write to is:

PC Disk Magazine
Problem Recovery
One Park Avenue
New York, N.Y. 10016

GAMES

JIG JAGS 2

By Ron Dubren & Associates

Special Requirements: None
Files Used: JIGJAGS.BAS
PIECES
PUZZLE.SMP
PUZZLE4
PUZZLE5
PUZZLE6

What do you get when you cross a jigsaw puzzle with a crossword puzzle? You get a JIG JAGS, *of course—a game which combines the spatial and verbal dexterity necessary to solve either of these types of puzzles. This clever word game is the second installment of a* PC Disk Magazine *regular feature. The next issue will also bring you three new puzzels for this challenging game created by Ron Dubren.*

THE BASICS

JIG JAGS is played with a 7-by-7 square crossword grid that has been divided into 16 pieces. The object of the game is to arrange those pieces on the blank 7-by-7 grid so that they fit together to form complete words both down and across. There are no crossword puzzle-type clues, so you must guess what the words are by looking at the letters only. Hence, in addition to resembling a crossword puzzle and a jigsaw puzzle, JIG JAGS is somewhat like an anagram. At the end of each timed game, you will be assigned a rank based on how long it took you and how much help you required.

START-UP

Load Advanced BASIC into your computer by putting a system disk into your default drive and typing:

BASICA ↵

Then place your work copy of the *PC Disk Magazine* diskette into your default drive and type:

RUN "JIGJAGS ↵

After the title screen you will see the main menu, called the "Options Screen." If you're playing *JIG JAGS* for the first time, press:

S

to see a sample game. (Be sure to keep your Caps Lock key toggled on when playing *JIG JAGS*.) The computer will play a sample game, showing the basic board and grid arrangement and the fit of the pieces. From the sample game you can press any key, at any time, to return to the Options Screen.

The "E" option on the Options Screen exits the *JIG JAGS* program and returns you to BASIC. Pressing:

↵

begins a game.

THERE ARE THREE PUZZLES AVAILABLE

To play *JIG JAGS,* you must first choose one of the three puzzles available in this issue. You will then be asked to choose a difficulty level. "Hard" is the beginning level. Only *JIG JAGS* pros should attempt the "Very Hard" level. Select one, press the Enter key, and plunge in.

THE GAME SCREEN

The program will first construct its game screen. In the center is the grid upon which you will try to arrange the pieces of the puzzle. In the "Hard" level of play, the location of the blocked-out squares (the JIG JAGS equivalent of the black squares of a crossword puzzle) will be shown on the grid. Note that there can be only one solid square per row or column. At the advanced level of play, the blocked-out squares exist, but their positions are not shown.

Spread around the sides and top of the game screen is the "Bank"—the 16 pieces of the fragmented puzzle. Each piece contains two to five squares, and the squares enclose either letters or crosshatching (to signify a blocked-out square). Note that the pieces are already aligned on their axes—unlike a jigsaw puzzle, you don't have to figure out which end is up.

You can move the cursor throughout the grid or the bank by using

the directional arrows on the numeric keypad. The Space Bar will toggle the cursor from the bank to the grid and back again.

To work on a particular piece in the bank, move the cursor to that piece. The squares of that piece will begin flashing, and the cursor will land on a specific square in each piece. This is called the "index square," and is used to "pin" a piece from the bank onto the grid.

PLAYING JIG JAGS

There are three procedures for moving pieces in *JIG JAGS*. To move a piece from the bank to the grid, move the cursor to that piece in the bank and press:

F1

The squares of that piece will stop flashing and turn into double lines, and the letters in the piece will flash instead. When a second cursor appears in the grid, move it to the square on the grid where you want to "pin" the piece's index square. (The first cursor, remaining in the chosen piece in the bank, reminds you which is the index square.) Once you have located the cursor in the proper grid square, press the F1 key again and the move will be made. A flashing duplicate of the piece will appear on the grid.

Once the move is complete, you may either use the arrow keys to move the cursor elsewhere on the grid or use the Space Bar to move it back to the bank. This movement will stop the flashing of the recently-moved piece. The piece will retain its double outline on the grid and in the bank, to set it apart from the unused pieces.

USE THE ARROWS TO MOVE THE CURSOR ON THE GRID

From now on, any time you move the cursor to any square of this piece on the grid or in the bank, the piece will flash in both locations. This is useful when the grid starts getting full and you want to know which piece contains a certain letter.

If you put a piece in the wrong place and wish to place it back in the bank, move the cursor to any square of the piece on the grid and press:

F2

The piece will disappear from the grid and reappear in the bank. The cursor will remain on the grid.

You may also move a piece from one location to another within the grid. Move the cursor to any square of the piece on the grid. The piece will flash on the grid and in the bank. Press:

F3

to make the grid piece revert to a single outline. Move the cursor to the point where you want the index square of the piece to appear, and press F3 again. The move will be made and the game will stop for a moment while the computer reformats the board.

If you try to move a piece to a location where it would exceed the boundaries of the grid or another piece, or if you try to move a piece with a blocked-out square to a position lacking a corresponding blocked-out square, you will receive an error message.

SCORING

You play against the clock in *JIG JAGS*. For every 15 minutes that you work on the puzzle, you drop one level in the skill rating scale, which runs from "Grand Master" (under 15 minutes to solve the

puzzle) to "Unranked Amateur" (over 150 minutes). If you want to check your progress at any point during the game, put the cursor in the bank and press:

T

Your elapsed time will appear at the bottom of the screen. If you have chosen to play at the "Very Hard" level, you get a bonus 15-minute period.

If you find that *JIG JAGS* is a little too tough for you, put the cursor in the bank and press:

C

to get a clue. The program will take one piece from the bank and correctly place it on the grid. You will not be able to move that piece once it is placed, and there is a scoring penalty: you will drop one skill level for every clue you request. In the "Very Hard" game you get one clue free.

YOU WILL DROP ONE SKILL LEVEL FOR EVERY CLUE YOU REQUEST

HELP

A list of the playing options (F1, F2, F3, and the Space Bar toggle) is displayed at the bottom of the game screen as you play. For an explanation of other commands, place the cursor in the bank and press:

H

to access a help menu. This menu will explain the "C" and "T" keys, and the keys used to end the game.

ENDING THE GAME

The following commands can only be executed when the cursor is in the bank. Press:

O

to cancel the current game and return you to the Options Screen. Press:

Q

to cancel your current game and exit the program to BASIC. Press:

S

to save the current game in a file on the program disk. (Since the *PC Disk Magazine* diskette is write-protected, this will work only if you have copied the program to another diskette.) The game will be called up again, at the same level of completion and elapsed time, when you next boot up the program and begin to play. If you no longer wish to complete that game, press:

O

The saved file will be deleted, and you can begin a new game.

EXPANDING THE CODE

When you list any of the BASIC programs in your *PC Disk Magazine*, you will find that the code is often quite dense, and there is a conspicuous scarcity of comments. This does not represent the latest standard in BASIC coding. We would have loved to space the code out in an eye-catching fashion, and document it thoroughly with comments.

Unfortunately, the line numbers and comments take up storage space, and storage space is a precious resource to us and our readers at *PC Disk Magazine*. So we were forced to economize on space in order to maximize our product offering. To unravel the BASIC code, we suggest that you put it in a form that you can edit with some system editor (EDLIN or WordStar, for example) and then expand the spacing. To do this, you must convert the program from its *PC Disk Magazine* format to a standard ASCII file, which an editor can manipulate. The procedure for performing this conversion is first to load the program from the *PC Disk Magazine* diskette into BASIC. Then, using the A option, save the program on another diskette in the form of an ASCII file. Specifically, first load advanced BASIC by typing:

BASICA ↵

then put the *PC Disk Magazine* diskette in the default drive. Type:

LOAD"*programname* ↵

where "programname" is the name of a BASIC program on the *PC Disk Magazine* diskette. Lastly, replace the *PC Disk Magazine* diskette with a work diskette and type:

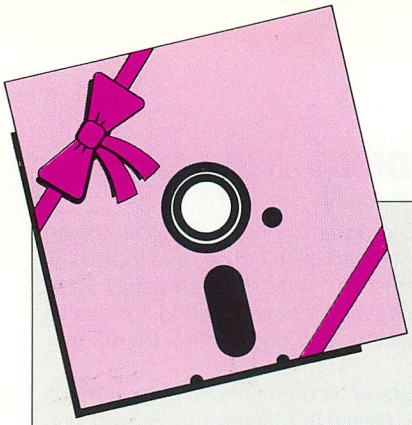
SAVE"*programname*", A ↵

An ASCII version of the program will now be written on this work diskette. You can then exit BASIC and call this program file into an editor. Use the editor to start a new line at every colon in the file. This will make the code much easier to read and the logic more apparent.

CALLING ALL PROGRAMMERS!!

If you have written software for the IBM PC that you feel would be of use or interest to others, *PC Disk Magazine* would like to know about it. Publishing is no longer just for poets and prosaists, but now it's for programmers too. Ask for our Software Submission Plan by writing to:

PC Disk Magazine
Author Submissions
One Park Avenue
New York, N.Y. 10016



EXPAND YOUR SOFTWARE LIBRARY

This issue of *PC Disk Magazine* has already increased your software library for your IBM personal computer. If you haven't subscribed to *PC Disk Magazine*, just think of how easy it will be to expand your library with future issues delivered to your door at substantial savings. And if you already are a subscriber, wouldn't you like to share your knowledge with a friend and get them a gift subscription to *PC Disk Magazine?*

Either way, you can save $60 with this introductory offer and give yourself or a friend up to 50 programs for as little as $3.00 each! *PC Disk Magazine* is the solution to the problem of building and maintaining an up-to-date library of tested software programs.

You will get up to 10 varied and ready-to-use programs and files in every issue—complete with an illustrated user manual—for the ultimate in ease, efficiency, and economy.

Every program in *PC Disk Magazine* is developed by experts, tested by the editors of *PC Disk Magazine* and published by the people who bring you *PC,* guaranteeing quality and satisfaction.

Use the enclosed card to subscribe today or to give a friend a gift. If the card is missing, please write to:
PC Disk Magazine, PO Box 5930, Cherry Hill, NJ 08034
For even faster service, call our toll-free number below and use your American Express, MasterCard or Visa . . . today!

CALL
1-800-526-0666
for FAST service
and get
6 issues of
PC Disk Magazine
for only $149
a savings of $60!

COMING UP

CURVE FITTER
One of five different curves is automatically fit to your data points to determine the relationship between them.

FILE MANAGER III
The final installment of this disk directory utility enables you to view or print your text files in whole or in part.

PRINT FORMATTER
Punch up your printouts with compressed, double width, emphasized, or double struck characters.

WAR IN THE FALKLANDS
As Commander of Her Majesty's Royal Navy Task Force, you issue orders to capture land sectors and retake the Falkland Islands!

JIG JAGS 3
Our continuation of the new trend in crossword puzzles. Construct three, new, valid crossword puzzles from jigsaw puzzle pieces.

WINE DATA 3
The conclusion of our series of wine rating reference files features the vintages of California.

BASIC TUTORIAL
Expand your knowledge of BASIC programming with this useful feature. Learn how to create enhanced graphics displays using the extended IBM character set, while exploring the inner workings of the PC.

LABEL MAKER
A mailing list management program for home or business use. Store thousands of addresses and print labels in any size or format.

FUNNELS & BUCKETS
Help your child learn practical math skills with this enjoyable tutorial. Presents basic math exercises in a challenging and entertaining game format.